A State Observer Based Methodology for Improving Control Schemes Employing Multiple Exogenous Feedforward Signals - An Undergraduate Research Thesis

Corey Marcus
Department of Mechanical and Aerospace Engineering
The Ohio State University
Columbus, Ohio, USA

Advisor: David Hoelzle
Department of Mechanical and Aerospace Engineering
The Ohio State University
Columbus, Ohio, USA

04/13/2018

Abstract

Feedback control provides the basis of many different control schemes. However, even high gain feedback may be insufficient for processes requiring high precision or non-causal behavior such as micro additive manufacturing or metrology. Exogenous feedforward inputs can be sometimes be used to provide a solution in these circumstances. These signals are carefully trained such that they produce the desired response in their target system. However, the efficacy of these signals can be greatly diminished when the systems they are applied to have different initial conditions from the ones for which the signals were designed. This problem is magnified when multiple feedforward inputs are applied sequentially. The subtype of Iterative Learning Control, Basis Task Iterative Learning Control (BTILC) involves creation of multiple exogenous feedforward signals which correspond to various learned behaviors. These signals are then applied sequentially in order to produce more complex system outputs without explicitly applying the learning algorithm to those outputs. This makes it a prime example of a control scheme which suffers from the decreased signal efficacy discussed previously. This manuscript first generates a novel algorithmic solution to these issues leveraging state information observed in the feedforward signal training process; called an Informed State Correction (ISC). Then, it presents experimental results which demonstrate a performance increase of approximately 70% in BTILC control schemes implementing an ISC. These results represent a significant increase in the efficacy of BTILC and its applicability to real-world control scenarios. Furthermore, the ISC has been posed such that it can be applied to any control scheme employing multiple exogenous feedforward signals, where it may provide similar performance benefits.

Contents

List of Figures

List of Tables

1 Introduction

The problem of control is an important one for many dynamic systems. Even the most precisely engineered system or process can fail to function without placing an equal amount of effort into controller design process. Many systems can be sufficiently controlled with simple feedback (FB) based Proportional Integral Derivative (PID) controllers. However, PID FB control may be inadequate for systems requiring very high levels of precision or non-causal behavior such as micro additive manufacturing or metrology. In such cases, feedforward (FF) control may provide a solution.

1.1 Problem Formulation

Consider the general FB loop which has been augmented with an exogenous FF input signal to improve upon output tracking performance in comparison to feedback control alone displayed in Figure 1. This system contains two individual subsystems, the plant and the controller. These are represented by **G** and **C**, respectively. This two input, single output system can be represented in transfer function form by

$$Y(s) = P(s)U(s) + T(s)R(s) \tag{1}$$

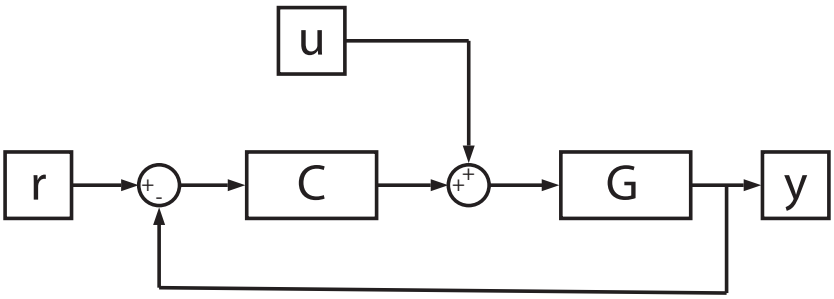

Figure 1: A FB control loop which has been augmented with an exogenous FF input.

where $R(s) = \mathscr{L}(r(t))$ is the Laplace Transform of the reference signal, $U(s) = \mathscr{L}(u(t))$ the Laplace Transform of the exogenous input signal, and $T(s) = \frac{C(s)G(s)}{1+C(s)G(s)}$ is the complementary sensitivity function, and $P(s) = \frac{G(s)}{1+C(s)G(s)}$ the transfer function between the exogenous input and the output. Error dynamics are written as

$$E(s) = -P(s)U(s) + \underbrace{S(s)R(s)}_{d} \tag{2}$$

where $S(s) = \frac{1}{1+C(s)G(s)}$ is the sensitivity function. Exogenous input signal $u(t)$ is reference specific. We will assume that via good feedback controller design that the error will be small and will act as a disturbance term d in Eqn. (2); the analysis herein will be focused on the system $P(s)$.

It will be assumed that the exogenous input signal is being accessed from a signal database for each specific reference signal or segment of a reference signal. Accordingly, the most natural framework for the our analysis will be a discrete-time state-space framework. **C** is presented in this framework as follows

$$\mathbf{C}: \begin{cases} x_C(k+1) = A_C x_C(k) + B_C u_C(k) \\ y_C(k) = C_C x_C(k) + D_G u_C(k) \end{cases} \tag{3}$$

where $x_C(k) \in \mathbb{R}^{n\times 1}$ is the state vector, $u_C(k) \in \mathbb{R}^{1\times 1}$ and represents the controller input, and $y_C(k) \in \mathbb{R}^{1\times 1}$ is the controller output. Additionally, $A_C \in \mathbb{R}^{n\times n}$, $B_C \in \mathbb{R}^{n\times 1}$, $C_C \in \mathbb{R}^{1\times n}$, $D_C \in \mathbb{R}^{1\times 1}$. **G** may be represented in a similar fashion as

$$\mathbf{G}: \begin{cases} x_G(k+1) = A_G x_G(k) + B_G u_G(k) \\ y_G(k) = C_G x_G(k) + D_G u_G(k) \end{cases} \tag{4}$$

where $x_G(k) \in \mathbb{R}^{n\times 1}$ is the state vector, $u_G(k) \in \mathbb{R}^{1\times 1}$ and contains the exogenous input and controller output, and $y_G(k) \in \mathbb{R}^{1\times 1}$ is the output. Additionally, $A_G \in \mathbb{R}^{n\times n}$, $B_G \in \mathbb{R}^{n\times 2}$, $C_G \in \mathbb{R}^{1\times n}$, $D_G \in \mathbb{R}^{1\times 1}$. In Section 2, the state corresponding to the output will be segregated from other states, thus motivating the use of the Observable Canonical Form, in which $y_G(k) = x_n(k)$. To elucidate this property, observable canonical form is given below [3].

$$A_G = \begin{bmatrix} 1 & 0 & \dots & 0 & -a_n \\ 0 & 1 & \dots & 0 & -a_{n-1} \\ \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & \dots & 1 & -a_1 \end{bmatrix}$$

$$B_G = \begin{bmatrix} b_n - a_n b_0 \\ b_{n-1} - a_{n-1} b_0 \\ \vdots \\ b_1 - a_1 b_0 \end{bmatrix}$$

$$C_G = \begin{bmatrix} 0 & 0 & \dots & 0 & 1 \end{bmatrix}$$

$$D_G = b_0$$

Referencing Figure 1 once again, the following relationships between **C** and **G** are readily apparent.

$$u_C(k) = r(k) - y_G(k)$$
$$u_G(k) = y_C(k) + u(k)$$

These relations can be leveraged to establish a new discrete time state space system, **H**, which represents the two-input, single output, closed-loop system shown in Figure 1. The state-vector for **H** is defined as the following

$$x = \begin{bmatrix} x_C \\ x_G \end{bmatrix}. \tag{5}$$

The input to **H** has the following form,

$$u = \begin{bmatrix} u_C \\ u_G \end{bmatrix}. \tag{6}$$

The state-space model of **H** has the following representation

$$\mathbf{H}: \begin{cases} x(k+1) = Ax(k) + Bu(k) \\ y(k) = Cx(k) + Du(k) \end{cases} \tag{7}$$

where

$$A = \begin{bmatrix} A_C & \underline{0} \\ \underline{0} & A_G \end{bmatrix}$$
$$B = \begin{bmatrix} B_C & \underline{0} \\ \underline{0} & B_G \end{bmatrix}$$
$$C = \begin{bmatrix} \underline{0} & \underline{0} \\ \underline{0} & C_G \end{bmatrix}$$
$$D = \begin{bmatrix} D_C & \underline{0} \\ \underline{0} & D_G \end{bmatrix}$$

where $\underline{0}$ is an appropriately sized null matrix.

The central idea to be presented in this manuscript centers around increasing the efficacy of these exogenous FF control schemes. Because the exogenous FF signals lack feedback, they operate in many ways like an open loop control scheme. Should the system performance begin to differ from the desired behavior, it is very possible for the

FF signals to begin to exacerbate the problem. By using a state estimator to record the desired system performance during FF signal generation, it is possible to introduce a FB control loop into the system which is capable of directly modifying the exogenous FF signal. The remaining sections of this manuscript seek to support and validate this central idea.

1.2 Iterative Learning Control

Experimental validation of the algorithm presented in Section 2 of this manuscript will involve an Iterative Learning Control (ILC) based control scheme. Bristow *et al* writes extensively about the various forms of ILC [1]. In short, ILC is a broad class of algorithms which use an iterative cycle to generate an input signal which learns from past inputs and the resultant errors. An ILC learning process is described in Figure 2. Note the addition of an iteration domain perpendicular to the page. The input signal is evolved between each iteration according to the algorithms update law such that the error signal of the system is reduced in magnitude with each iteration. Parallel architecture ILC uses an ILC controller in parallel with the existing feedback controller; much like the architecture in Figure 1. The update law is used to generate exogenous FF signals such as those used in Figure 1. This results in the plant input becoming the sum of the feedback controller output and the exogenous FF signal. ILC has the capability to designed as an non-causal system because the signal generation algorithm is executed off-line where all time domain data from the previous iteration is already available.

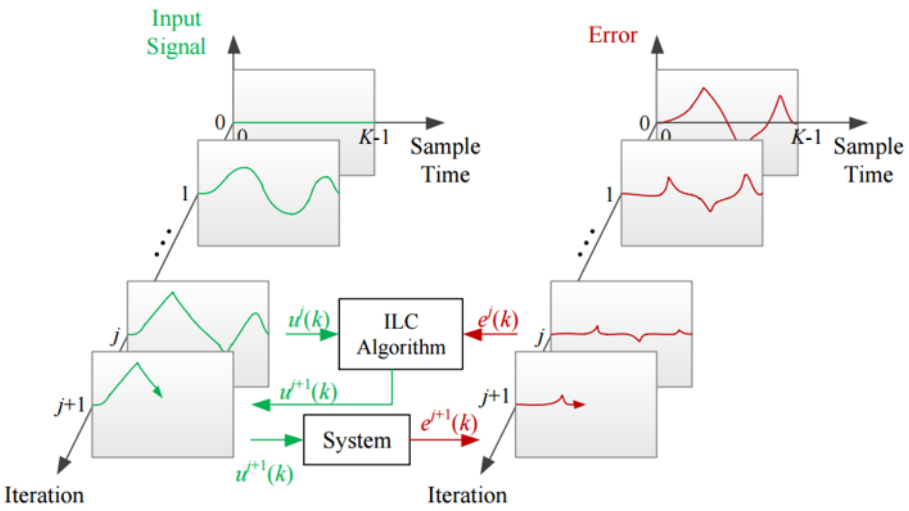

Figure 2: An illustration of the ILC process [1].

The efficacy of ILC drops rapidly if the desired system behavior deviates from the system behavior which was used in the learning process [1]. One particular subset of ILC, Basis Task ILC (BTILC), is an attempt to eliminate this drawback to ILC. The BTILC process is outlined in Figure 3 [2], [4]. BTILC aims to create a library of exogenous FF

signals which pertain to various basic tasks. These basic tasks now form a sort of pseudo-basis, and can be combined to produce more complex system behaviors. The goal is to enable these complex behaviors to be executed with the performance increases of ILC without using the ILC training process to create inputs for these complex behaviors explicitly. To accomplish this, a continuous trajectory containing each of the desired basis tasks is assembled. This trajectory is referred to as the *Training Set*. ILC is used to create a continuous input which causes the system to track the training set. The resulting input is then divided into segments such that each basis task is associated with a segment of the input signal based on the time indicies at which the basis task was executed. These input signals are used to build up a *Basis Signal Library*. If a user desires execution of a certain sequence of basis tasks, the corresponding input signals are pulled from the library and arranged into the *Operation Set*. Applying the operation set to the system then results in the user's desired basis task sequence being executed.

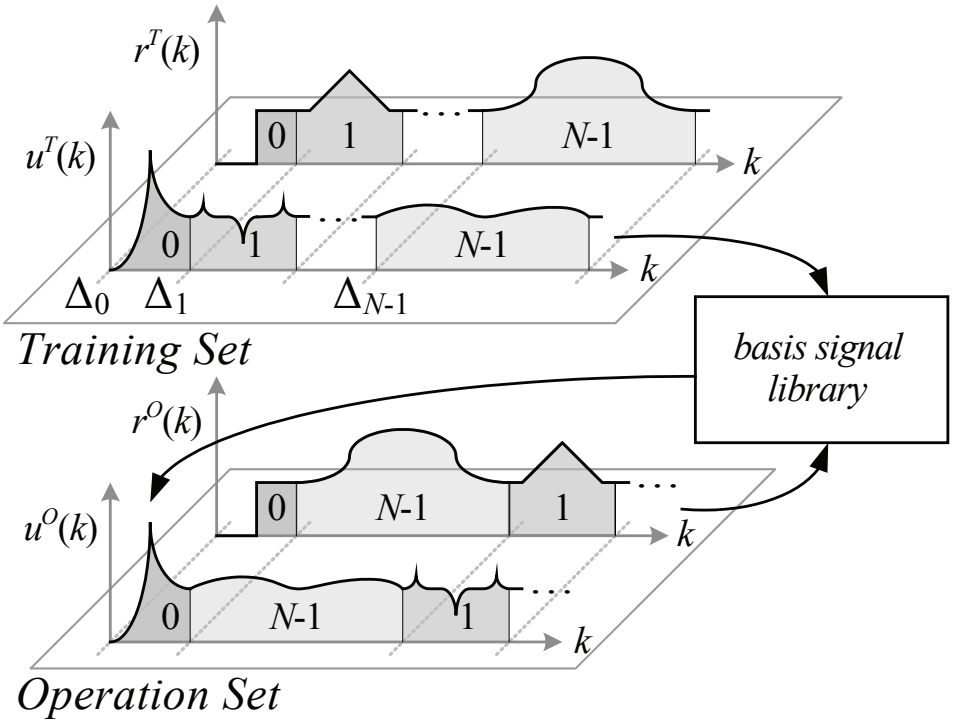

Figure 3: An illustration of the BTILC process [2].

To ensure clarity of communication in regards to the ideas discussed above, we introduce definitions of these terms as they which will be leveraged heavily in the remainder of the manuscript. First we define *Training*:

Definition. Training: The algorithmic process for creation of each $u_i^T(k)$ such that the output of **G** tracks each corresponding $r_i^T(k)$.

Second, we introduce an operation known as *recombination* which is defined as:

Definition. Recombination: An operation which pulls input signals from the basis signal library and arranges them

sequentially. The signals can be recombined in any order and recombination does not preclude repetition or exclusion of signals. In Figure 3, recombination is used to produce the reference and input in the operation set, $r^O(k)$ and $u^O(k)$.

Finally, we introduce a process known as *operation*:

Definition. Operation: The application of a recombined sequence $u^O(k)$ to **G**. The intent of operation is for the output of **G** to track the recombination $r^O(k)$ which has same recombination as $u^O(k)$.

We now introduce two superscripts, T and O, representing *training* and *operation*, respectively.

The rest of the manuscript will be formatted as follows: Section 2 details the main contribution of this paper: the definition of an Informed State Correction Framework (ISC) to modify signals extracted from a database of exogenous FF based on current state, $x(k)$, information. ISC is applied to a two-axis robotic positioning system: Section 3 provides the experimental setup; including generation of exogenous FF signals via a BTILC methodology. Section 4 provide the experimental results of ISC application. Section 5 provides a discussion of the results and conclusions on the contributions of this work and future directions.

2 Informed State Correction Framework

This section presents the primary contribution of this work: the definition and derivation of an additional input to systems employing exogenous FF signals such that the system state vector tracks the trajectory for which the FF input was designed.

2.1 Assumptions

Several assumptions are made in the proceeding sections regarding the plant, **G**.

1. **G** is either stable or stabilizeable.
2. **G** has full state observability.

2.2 The Informed State Correction Algorithm

Consider a set of reference trajectories, **R**, for a system which is in fact a composite of N smaller trajectories, r_i, which are piecewise continuous and applied to a system, **G**, sequentially.

$$\mathbf{R} = [r_0(k), r_1(k), \ldots, r_{N-1}(k)] \in \mathbb{R}^{k_{\max} \times N}. \tag{8}$$

The transition between each individual reference signal segment $r_i(k)$, occurs on either a time or event based schedule. $N \in \mathbb{Z}$ is the number of reference signal segments applied and each $r_i(k)$ is defined on the domain $k \in [0, \Delta_{i+1} - \Delta_i - 1]$ and each Δ_i is the ending time index of $u_{i-1}(k)$. The signal transition time indices, Δ_i, are given by the following schedule:

$$\mathbf{\Delta} = [\Delta_0, \Delta_1, \ldots, \Delta_{N-1}, \Delta_N] \in \mathbb{R}^{1 \times N+1}. \tag{9}$$

The reference signal applied to the system may be expressed as

$$r(k) = \sum_{i=0}^{N-1} r_i(k - \Delta_i)(\mathfrak{s}(k - \delta_i) - \mathfrak{s}(k - \Delta_{i+1} - 1)). \tag{10}$$

The subscript i denotes the reference signal index, $k \in [\Delta_i, \Delta_{i+1} - 1]$, and the signal $\mathfrak{s}(k)$ is the unit step function where the difference of two step functions, offset in time, turn on and off a signal.

Similarly, a set of exogenous FF signals, **U** to be are to be applied sequentially to **G** as follows:

$$\mathbf{U} = [u_0(k), u_1(k), \ldots, u_{N-1}(k)] \in \mathbb{R}^{k_{\max} \times N}. \tag{11}$$

The exogenous signal applied to the system may be expressed as

$$u(k) = \sum_{i=0}^{N-1} u_i(k-\Delta_i)(\mathfrak{s}(k-\delta_i) - \mathfrak{s}(k-\Delta_{i+1}-1)). \tag{12}$$

The subscript i denotes the exogenous signal index, $k \in [\Delta_i, \Delta_{i+1}-1]$, and the signal $\mathfrak{s}(k)$ is the unit step function where the difference of two step functions, offset in time, turn on and off a signal. Note the presence of both $u(k)$ and $r(k)$ in Figure 4 as $u^T(k)$, $r^T(k)$, and their recombinations $u^O(k)$, and $r^O(k)$.

These signals have been well designed with an training process such that the output of $\mathbf{G}$ tracks a set of reference signals given by $\mathbf{R}$. Additionally, during application of $\mathbf{U}$ as it is designed in training the state vector of $\mathbf{G}$, $x_G \in \mathbb{R}^{n\times 1}$, tracks a set of signals

$$\mathbf{X} = [x_0(k), x_1(k), \ldots, x_{N-1}(k)] \in \mathbb{R}^{k_{\max}\times n\times N}. \tag{13}$$

A sequence of reference signals (8), inputs (11), and state values (13) and will be represented in the compact form $\mathbf{U}_\Delta$, $\mathbf{R}_\Delta$, and $\mathbf{X}_\Delta$, respectively. If any of these sequences were to be formed in training, they would be given the superscript T. For example, a $\mathbf{U}_\Delta$ would become $\mathbf{U}_\Delta^T$.

Figure 4 represents a modification to Figure 3 to represent the addition of the systems state response to the training process. $\mathbf{X}_\Delta^T$ is recorded with a state estimator during training and stored into a library alongside $\mathbf{U}_\Delta^T$ and $\mathbf{R}_\Delta^T$ for future use.

During the aforementioned future use, the user may desire a recombination of the order of $\mathbf{R}_\Delta^T$ during operation. This recombination is termed, $\mathbf{R}_\Delta^O$. Performing an identical recombination on $\mathbf{U}_\Delta^T$ to form $\mathbf{U}_\Delta^O$ allows the exogenous signals corresponding to each $r_i(k)$ to be applied at the time their execution is desired. At this point, two factors will contribute to decreased efficacy of each individual portion of $\mathbf{U}_\Delta^O$ when compared to $\mathbf{U}_\Delta^T$.

1. The training which formed $\mathbf{U}_\Delta^T$ assumed that the initial conditions of each individual u_{i+1}^T were equal to the final conditions of each u_i^T. Recombination has made this assumption invalid as sequential signals in training can no longer be guaranteed to be sequential in operation.

2. All real systems are non-linear, time-varying, and display stochastic properties. This implies that repeated application of identical input signals will produce an non-identical output. Even without recombination, this fact decreases the efficacy of each individual u_i^T in operation.

System performance can be improved with one critical insight. The same recombination which formed $\mathbf{U}_\Delta^O$ can be used to form $\mathbf{X}_\Delta^O$. Forming a negative feedback control loop in which $\mathbf{X}_\Delta^O$ is subtracted from the current system state, $\tilde{x}_G$, allows for calculation of an additional input to $\mathbf{G}$, $\delta(k)$, which will translate $\tilde{x}$ to $\mathbf{X}_\Delta^O$. This $\delta(k)$ is henceforth termed

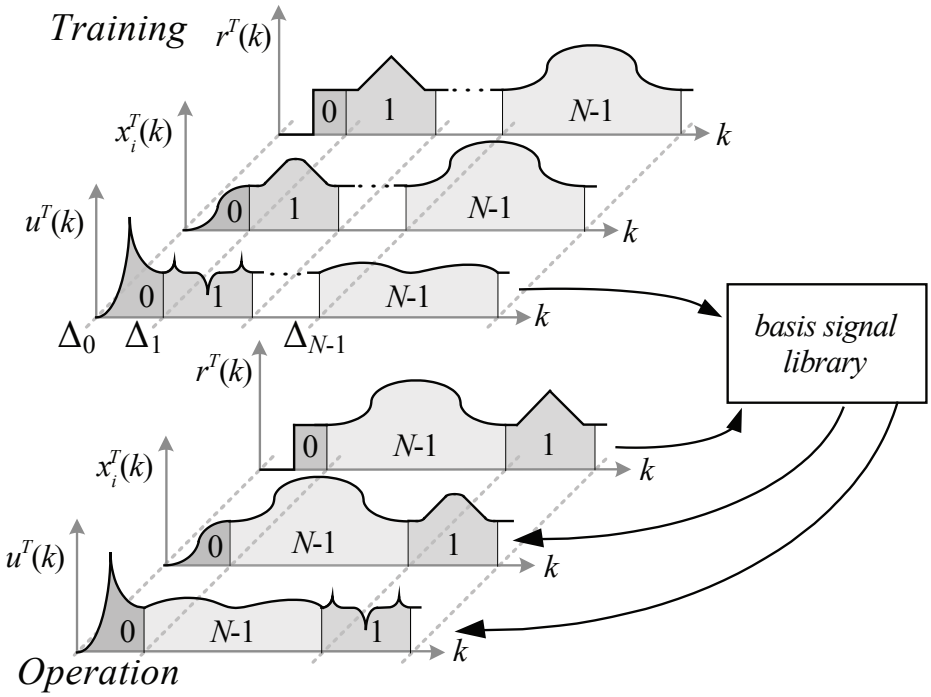

Figure 4: An outline of the recombination process (adapted from [2]).

an *Informed State Correction* (ISC). The ISC is applied to a system in the manner shown in Figure 5. The control scheme retains its PID FB loop and exogenous FF input. An additional FB loop has been created incorporating a state estimator and the state information gathered in training. The difference calculated in this loop is used to calculate the ISC. The input to the plant is now the sum of the PID controller output, exogenous input, and ISC.

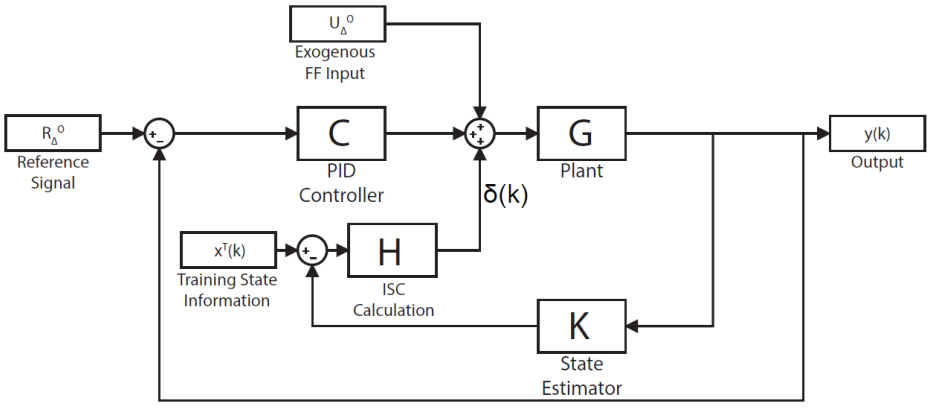

Figure 5: The Block Diagram Describing ISC Application

2.3 $\delta(k)$ Derivation

Equation (4) can be used to model the true performance of $\mathbf{G}$, $\tilde{x}_G$, during application of $\mathbf{U}_\Delta^O$:

$$\begin{aligned}\tilde{x}_G(k+1) &= A_G\tilde{x}_G(k) + B_G u_G(k)\\ \tilde{y}_G(k) &= C_G\tilde{x}_G(k) + D_G u_G(k)\end{aligned} \tag{14}$$

For the reasons outlined above, $\tilde{x}_G$ will not precisely track the desired plant behavior, $\mathbf{X}_\Delta^O$. This desired plant behavior displays the following dynamics:

$$\mathbf{X}_\Delta^O(k+1) = A_G\mathbf{X}_\Delta^O(k) + B_G\mathbf{U}_\Delta^O(k) \tag{15}$$

The output of $\mathbf{G}$ and be controlled to track the output of $\mathbf{X}_\Delta^O$ by providing an additional input, $\delta(k)$, to $\mathbf{G}$. $\delta(k)$ will be calculated on-line during operation such that

$$\tilde{x}_G(k+1) = \mathbf{X}_\Delta^O(k+1) \tag{16}$$

The next series of equations present the derivation of $\delta(k)$ from Equation (16). With the inclusion of $\delta(k)$ and substitution using Equation (14) and Equation (15):

$$\begin{aligned}A_G\tilde{x}_G(k) + B_G(\mathbf{U}_\Delta^O(k) + \delta(k)) =& A_G\mathbf{X}_\Delta^O(k) + B_G\mathbf{U}_\Delta^O(k)\\ B_G\delta(k) =& A_G\mathbf{X}_\Delta^O(k) - A_G\tilde{x}(k) + B_G\mathbf{U}_\Delta^O(k) - B_G\mathbf{U}_\Delta^O(k)\\ B_G\delta(k) =& A_G(\mathbf{X}_\Delta^O(k) - \mathbf{X}_\Delta^T(k))\\ \delta(k) =& (B_G'B_G)^{-1}B_G'A_G(\mathbf{X}_\Delta^O(k) - \tilde{x}_G(k))\\ \delta(k) =& B_G^+A_G(\mathbf{X}_\Delta^O(k) - \tilde{x}_G(k))\end{aligned} \tag{17}$$

where $B_G^+ = (B_G'B_G)^{-1}B_G'$, and is the left pseudo-inverse of B_G.

2.4 Application Notes

A user may wish to condition $\delta(k)$ such that deviations within certain states produce stronger reactions or to control the amount of effort applied by the ISC. This can be done with a simple gain;

$$\boldsymbol{\alpha} = \text{diag}\left[\alpha_1, \alpha_2, \ldots, \alpha_n\right]$$

Applying $\boldsymbol{\alpha}$ to Equation 17 creates a final expression for $\delta(k)$.

$$\delta(k) = (B'B)^{-1}B'A\boldsymbol{\alpha}(\mathbf{X}_{\Delta}^{O}(k) - \tilde{x}(k)) \tag{18}$$

There is one caveat in creation of $\mathbf{X}_{\Delta}^{O}$ for systems which x have a position control objective. Consider a non-causal process which has been used to generate the system inputs such that the output of $\mathbf{G}$, a position control objective, executes a a variety of step inputs as shown in Figure 6. The figure shows the creation of seven different tasks which make up the three steps and a steady state dwell. The steady state dwell forms Task 1, while the remaining tasks are used to form the step behaviors. Each step is composed of two tasks. The first task controls the rise behavior in step execution, while the second controls the settling behavior.

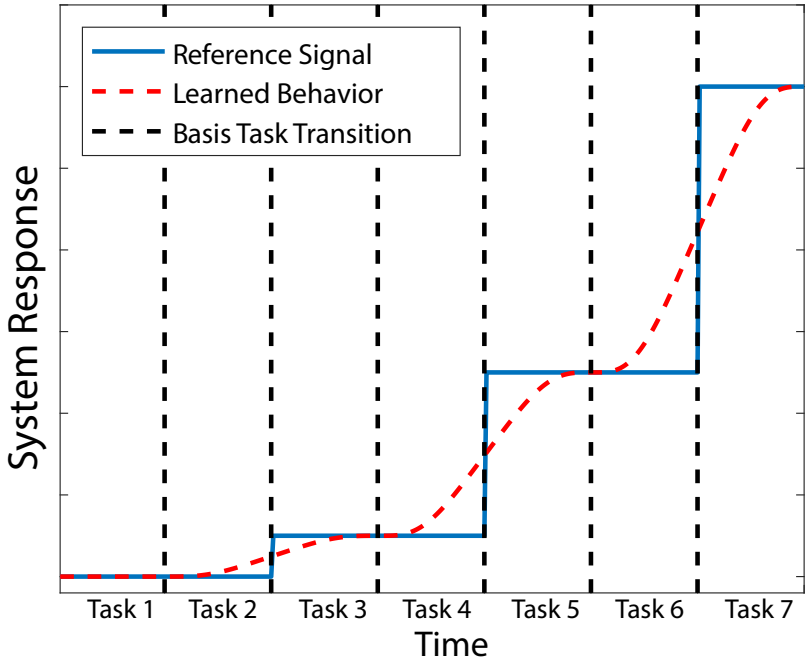

Figure 6: The results of a theoretical learning process

A user may desire the execution of the task sequence $[1,4,5,1]$ in operation. Recombination of $\mathbf{R}_{\Delta}^{T}$ and $\mathbf{X}_{\Delta}^{T}$ to produce $\mathbf{R}_{\Delta}^{O}$ and $\mathbf{X}_{\Delta}^{O}$ would result in the behavior shown in Figure 7. Clearly the learned behavior signals for the final three tasks do not agree with the reference signal. This has occurred because the signals were learned at a different position in training from their application in operation.

The solution to this problem is shown in Figure 8. Observable Canonical Form can be leveraged to ensure that the state $x_{i,n}(k)$ is actually the system's output. For each individual task, $x_{i,n}(k)$ is linearly translated such that $x_{i,n}(0) = r_i(0)$. It is now implied that any recombination used to create $\mathbf{X}_{\Delta}^{O}$ in which x contains a position control objective also includes the aforementioned translations.

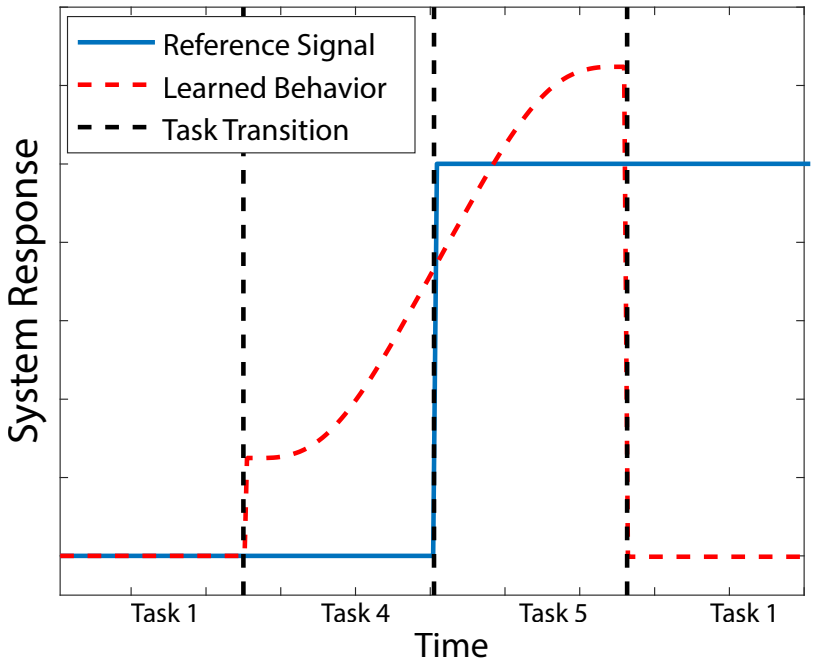

Figure 7: The misalignment of recombinations including spatial variables

3 Experimental Validation - Setup

3.1 The mPS

Experimental validation of the ISC was performed on an micro-scale Positioning System (mPS); pictured in Figure 9. The mPS is a precision robotic system containing three orthogonal motion stages. In order to reduce the scope of the problem, only the horizontal motion stages, x and y, were considered for this study.

An experimental bode plot was generated for each axis of the mPS by recording its steady state response to a sinusoidal input of various frequencies under unity feedback control. From these plots, plant models G_x and G_y were heuristically created such that they sufficiently matched the experimental performance of the mPS. These results are outlined in Figures 11a and 13a.

G_x and G_y were used as simulation models in the creation of two PID controllers, C_x and C_y which were used to form the feedback control loop for the mPS. The PID gains of the two controllers were heuristically tuned in simulation to produce a step response in each axis with low overshoot and a quick rise time.

A Kalman Filter [5] was used as a state estimator in order to provide full state information of the plant during operation. The process and sensor noise covariances of the Kalman Filter were determined using the Autocovariance Least-Squares Technique [6].

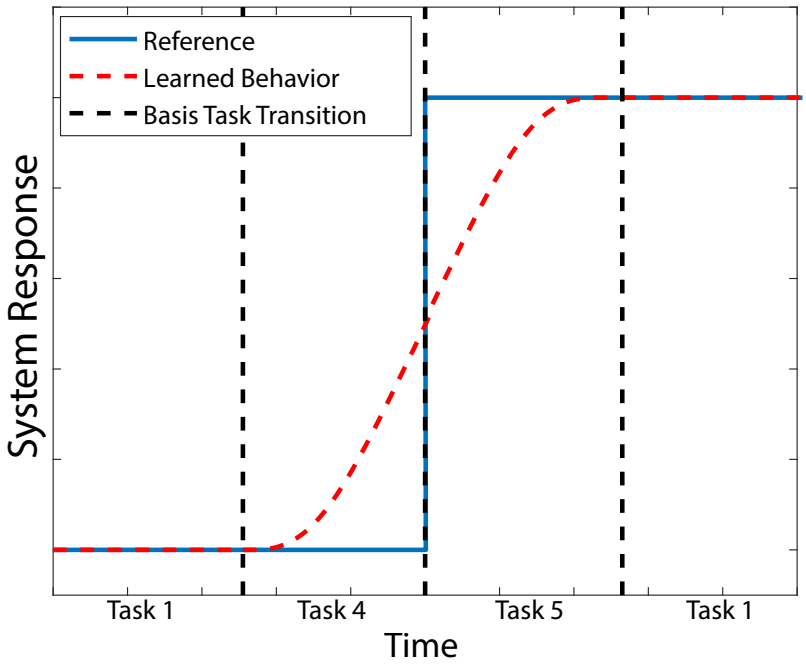

Figure 8: The final translation of the recombination such that the learned behavior tracks the reference signal

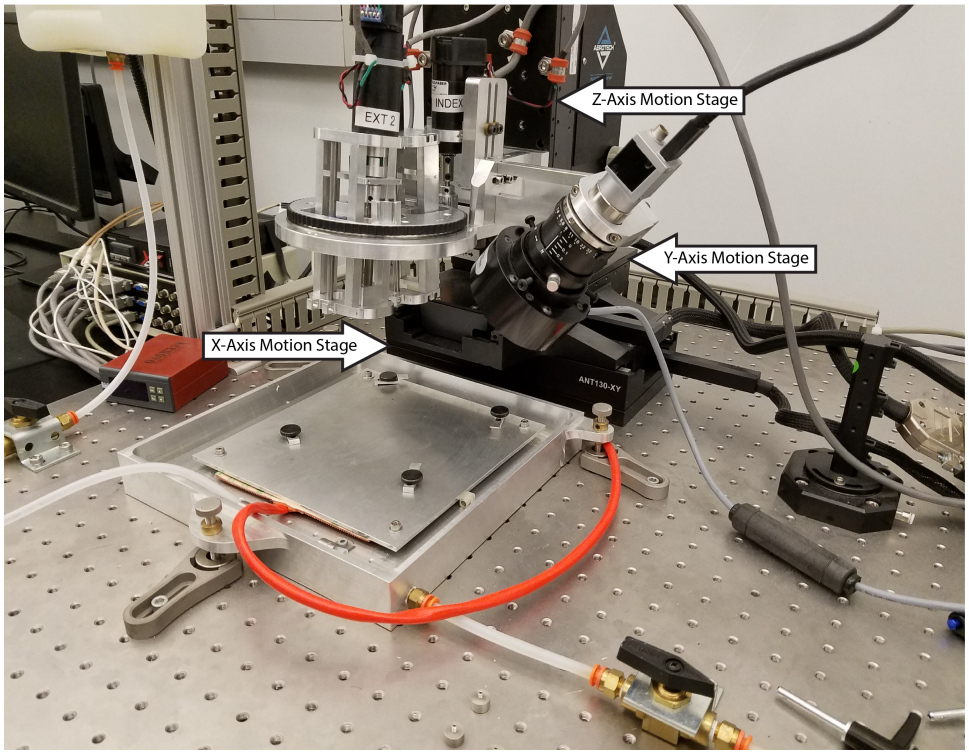

Figure 9: The micro-scale Positioning System (mPS)

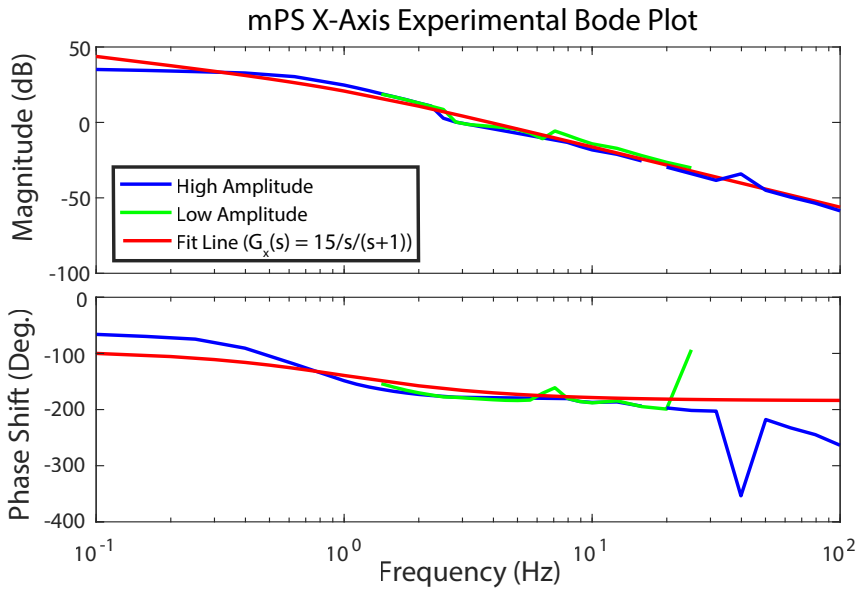

Figure 10: The Experimental X-Axis Bode Plot of the mPS.
Low and high amplitude refers to the input amplitude of the reference signal. High amplitudes (14V Pk-Pk) were used at low frequencies to negate the nonlinear effects of friction at low velocities. Lower amplitudes (4V Pk-Pk) were used at mid range frequencies to avoid damage to the motion stages at resonant frequencies of the unity feedback control scheme. High amplitude inputs (14V Pk-Pk) were again used at higher frequencies in order to increase the signal to noise ratio as the output of the system began to experience significant attenuation.

3.2 Exogenous FF Signal Generation

Exogenous FF signals were generated with an ILC training process. The PD learning algorithm shown in Equation 19 was used for ILC training [7]. The proportional and derivative learning gains, k_p and k_d, along with the non-causal Gaussian Filter, Q, of the algorithm were heuristically tuned to produce a monotonically decreasing error signal in the iteration domain during training.

$$u_{j+1}(k) = Q[u_j(k) + k_p e_j(k+1) + k_d[e_j(k+1) - e_j(k)]] \tag{19}$$

A set of basis tasks (BT) was created which includes accelerations, turns, and linear travels in each direction. These tasks were combined into one continuous trajectory, shown in Figure 14. Figure 15 shows the error signal evolution over the course of the training process.

3.3 Testing Conditions

The ILC input and Kalman filter output signals were then gathered into a library of BT input and output signals. The signals contained within this library were then recombined to produce three new trajectories. These trajectories

are shown below in Figure 16.

The mPS is asked to execute each trajectory under four testing conditions. Each testing condition is repeated 10 times for a total of 40 trials. The testing conditions are outlined as follows

1. Feedback Control - Using only the feedback control system. This control scheme is expected to provide the poorest tracking performance.

2. BTILC - Using input signal library generated in the Basis Task training as outlined in 3

3. BTILC + ISC - Using the input signal library generated in training Basis Task training along with the an ISC based on the state trajectory data which was gathered in 3.

4. ILC - using the learning algorithm directly to produce a continuous input without basis tasks. This control scheme is expected to provide the highest level of tracking performance.

3.4 Primary Data Analysis

The primary metric by which each trial will be analyzed is its error signal, $e(k)$. The error signal is defined as the overall spatial deviation from the desired trajectory at a given time instant. This can be easily derived with an

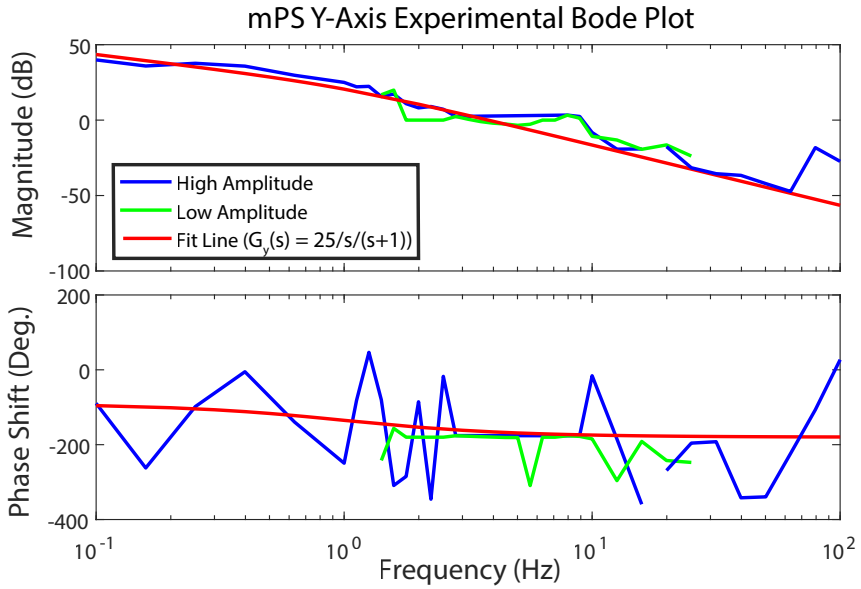

Figure 12: The Experimental Y-Axis Bode Plot of the mPS.
High amplitudes (14V Pk-Pk) were used at low frequencies to negate the nonlinear effects of friction at low velocities. Lower amplitudes (4V Pk-Pk) were used at mid range frequencies to avoid damage to the motion stages at resonant frequencies of the unity feedback control scheme. High amplitude inputs (14V Pk-Pk) were again used at higher frequencies in order to increase the signal to noise ratio as the output of the system began to experience significant attenuation.

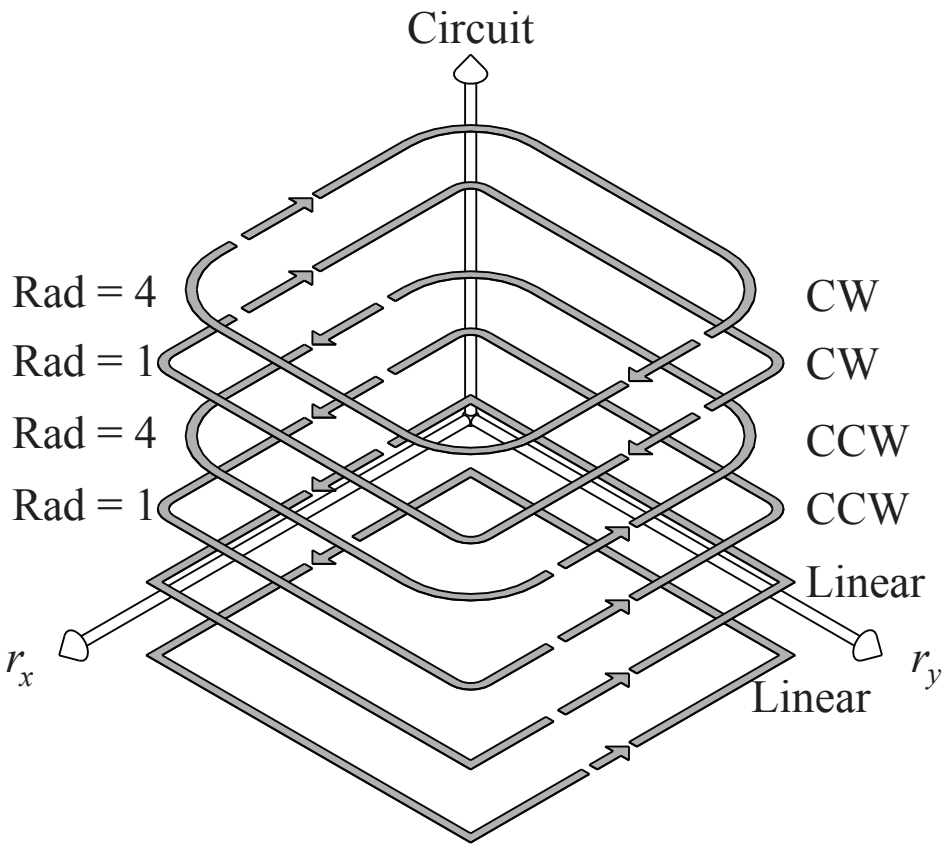

Figure 14: The learning Circuit for Basis Task Learning [2]

Euclidean Norm of the error in the x-axis and y-axis, $e_x(k)$ and $e_y(k)$, respectively.

$$e(k) = \sqrt{e_x(k)^2 + e_y(k)^2} \tag{20}$$

where

$$e_x(k) = r_x(k) - y_x(k) \quad \& \quad e_y(k) = r_y(k) - y_y(k).$$

To provide an metric of evaluation over a given time range the Root Mean Square (RMS) of the error signal was considered. The RMS of $e(k)$ is given by:

$$RMS(e(k)) = \sqrt{\sum e(k)^2} \tag{21}$$

To provide relative comparisons between trajectories, the RMS of each signal was normalized by highest performing execution of ILC (test condition 4) for a given trial. The highest performing execution of ILC is defined as the one with the lowest $RMS(e(k))$. This produces a quantity defined as the RMS Ratio (RMSR):

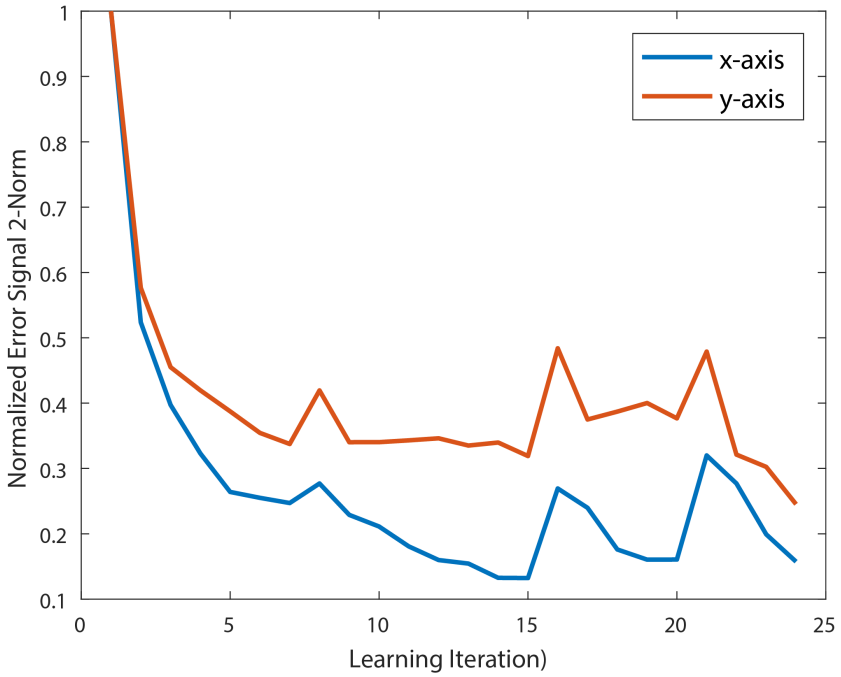

Figure 15: The Error Signal Evolution During the BTILC Training Process

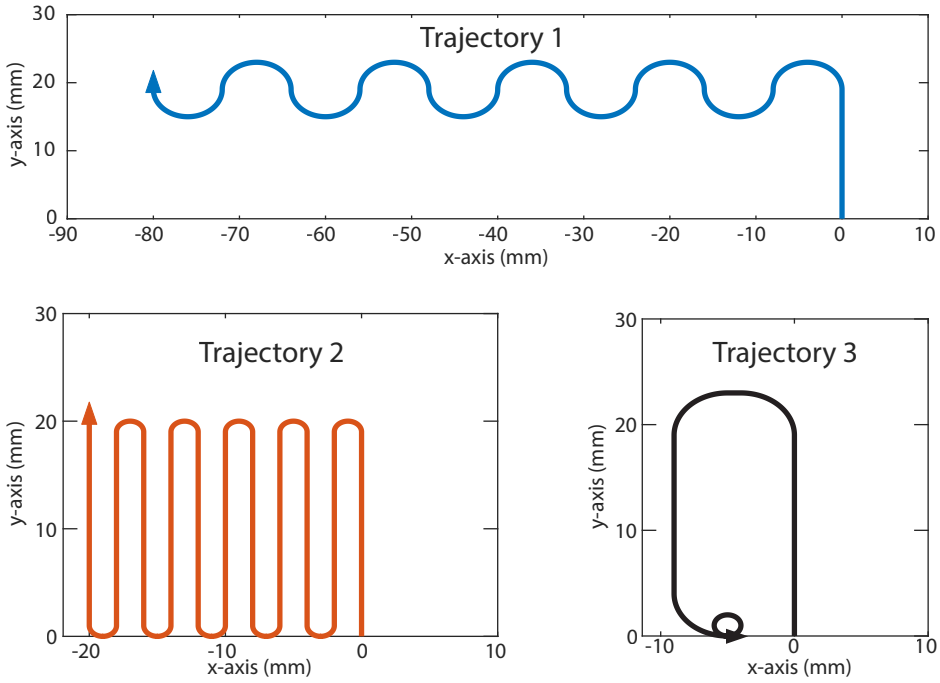

Figure 16: The Three Trajectories Used for ISC Validation

$$RMSR(e(k)) = \frac{RMS(e(k))}{min(RMS(e_{ILC}(k)))} \tag{22}$$

3.5 Secondary Data Analysis

Two forms of statistical analysis were performed in order to quantify the statistical significance of the results gathered. The first was a one-way ANOVA test which sought to determine if their was a statistically significant ($P < 0.05$) difference between each control scheme within a given trajectory. The null and alternative hypothesis for the ANOVA test are presented as

Null: The mean RMSR is the same for the FB, BTILC, and BTILC + ISC controllers.

Alternative: The mean RMSR is different for each of the FB, BTILC, and BTILC + ISC controllers.

The second form of statistical analysis performed is the Tukey Honest Significant Difference (HSD) test. It's purpose is to establish that there are statistically significant performance differences between each pair of controllers (FB/BTILC, FB/BTILC + ISC, BTILC/BTILC + ISC) in a given trajectory.

4 Experimental Validation - Results

4.1 Trajectory 1

The tracking performance of the mPS is shown in 17 for each operating condition. The view window has been constricted to the behavior in one turn which is indicative of system performance overall. The system is attempting to track the reference signal as it moves from right to left across the figure. Feedback control shows the poorest performance, displaying high levels of overshoot. All three forms of ILC based control exhibit undesirable oscillatory behavior. BTILC displays poorer tracking performance than BTILC + ISC; which in turn displays poorer performance than ILC.

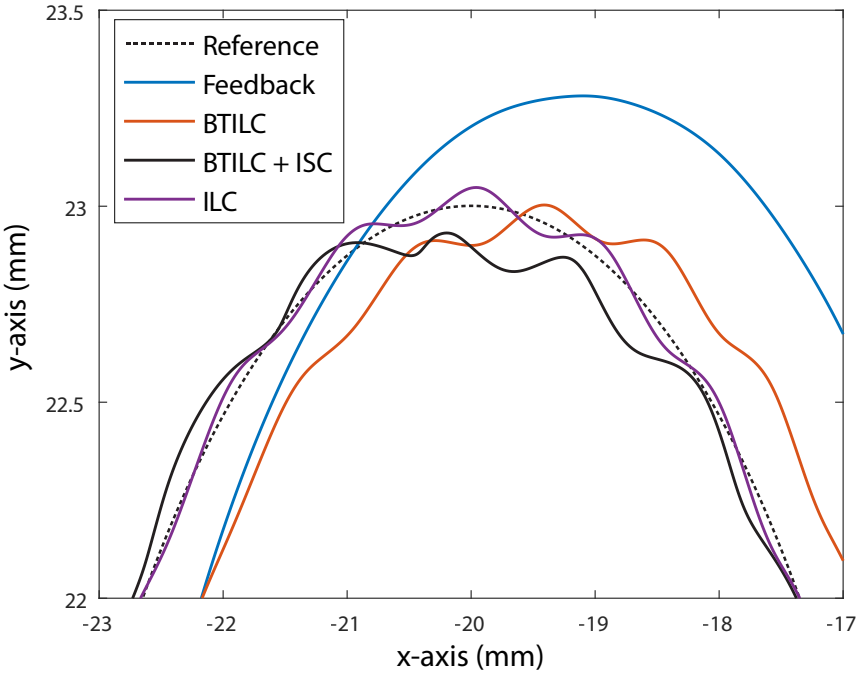

Figure 17: Tracking performance of Trajectory 1 during a chosen turn.

To further illuminate the differing levels of performance in the control schemes, the absolute error of the system is shown in Figure 18 for each operating condition. As indicated in Figure 17, Feedback control clearly has the largest error signals on average. Feedback control is followed by BTILC, BTILC + ISC, and ILC in terms of descending error signal magnitude.

4.2 Trajectory 2

In order to highlight the steady state performance of each control scheme, the straight line performance of each scheme is displayed in Figure 19. During the target time interval, the controllers are attempting to track a constant

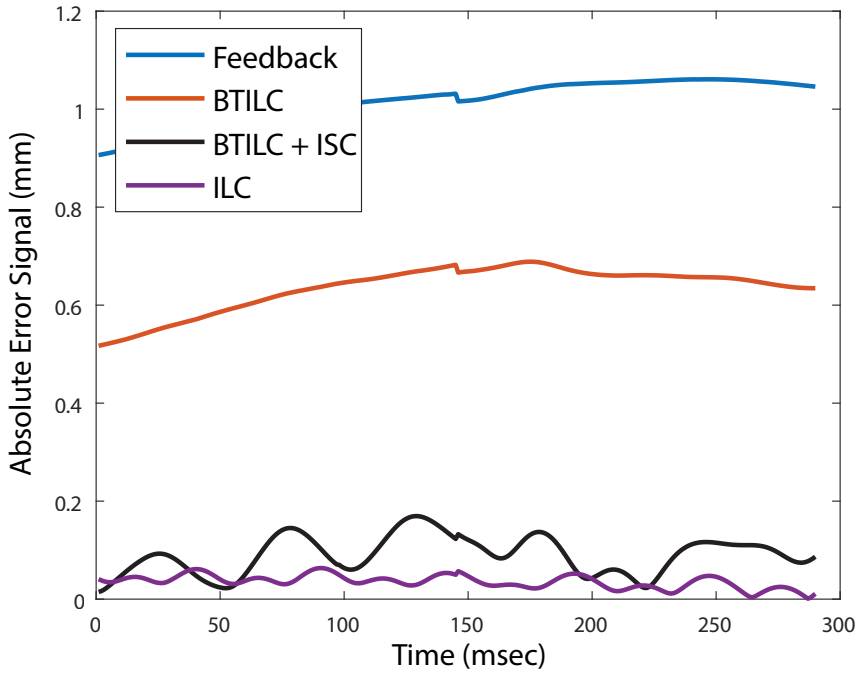

Figure 18: System error during execution of a chosen turn in Trajectory 1.

velocity travel in the positive y-axis. Feedback control displays some slowly collapsing steady state error in the x-axis. This error is a residual from the x-axis travel which occurred in the 180 degree turn immediately prior to the chosen sample window. BTILC also displays a relatively equivalent steady state error in the x-axis; however, this error does not display signs of collapse. BTILC + ISC displays tracking behavior which oscillates about the reference trajectory with a lower overall error magnitude than BTILC or feedback control. Finally, ILC displays the highest levels of tracking performance under steady state conditions. It shows no visible oscillations and the lowest apparent error magnitude.

4.3 Trajectory 3

Trajectory 3 contains the most challenging segment which the mPS was asked to perform; a 10 millimeter diameter circle at a tangential velocity of 20 millimeters per second. The tracking performance of each control scheme is shown in Figure 20. The results during this trajectory follow the same trends which have been shown in previous trajectories. Feedback control displays the poorest results, followed by BTILC, BTILC + ISC, and ILC in order of increasing performance. All three learning control schemes display unfavorable oscillatory behavior. The absolute time domain error signals are shown in Figure 21 and support the observations described above.

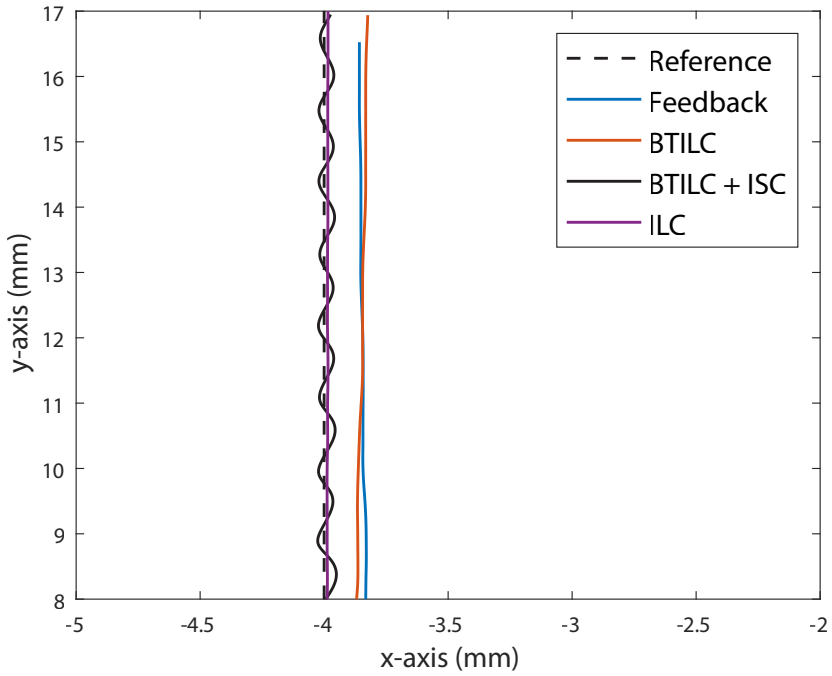

Figure 19: Tracking performance of Trajectory 2 during a straight line segment.

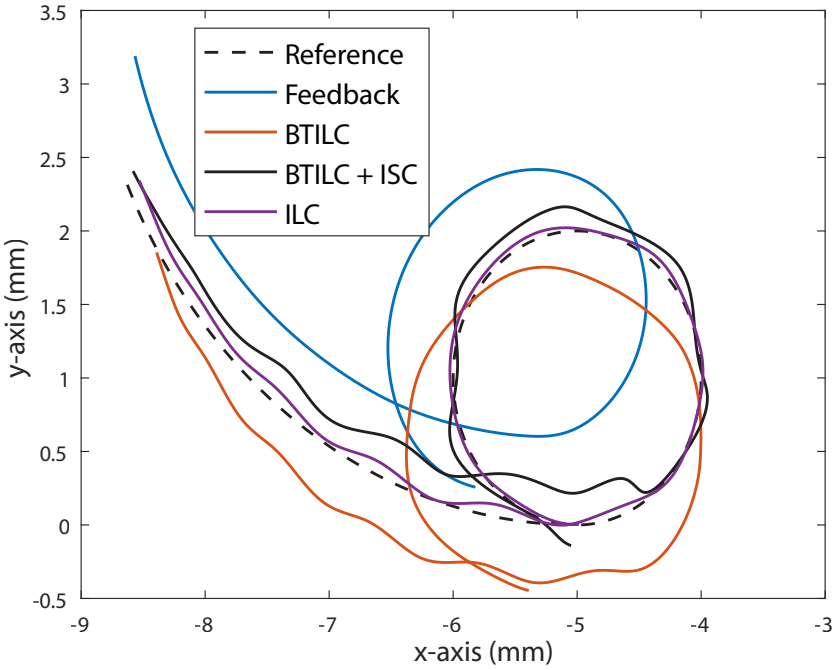

Figure 20: Tracking performance of Trajectory 3 during the tight loop

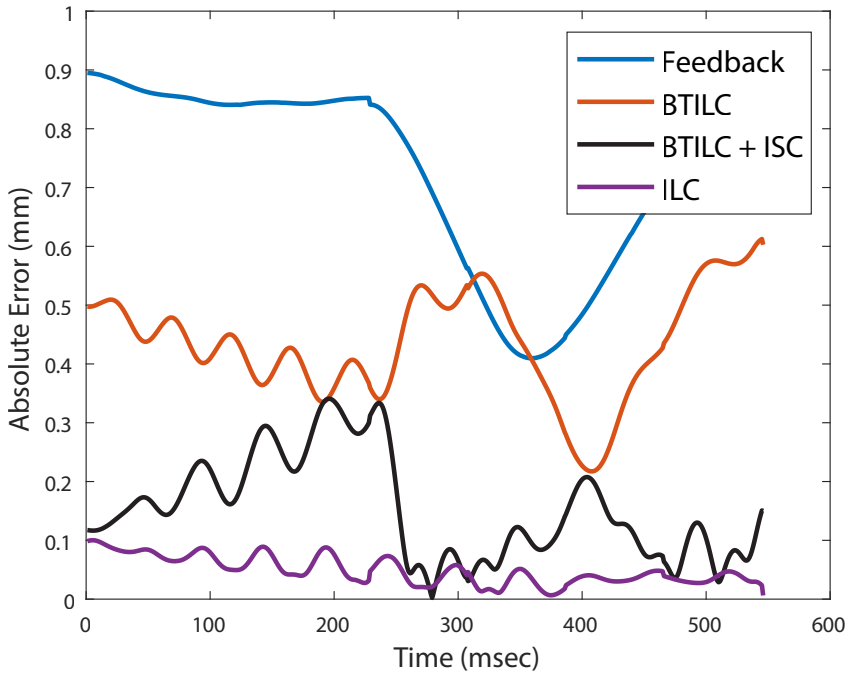

Figure 21: System error during execution of the tight loop in Trajectory 3.

4.4 Holistic Results

As mentioned previously, each of the three trajectories was executed under each control scheme 10 times. This allowed the consistency of the various methodologies to be analyzed. The primary metric through which results are displayed is the RMSR, 22. The average RMSR is displayed along with the standard deviation of the set in Figure 22. As suggested in the earlier results, BTILC + ISC displays a lower average RSMR than both BTILC and feedback control. The mean RMSR values used to create Figure 22 are tabulated in Table 1

Table 1: The mean RSMR values for each control scheme

Trajectory	FB	BTILC	BTILC + ISC
1	25.18	19.08	3.90
2	15.27	12.14	3.46
3	13.05	8.01	3.22

The results of the one-way ANOVA and Tukey's HSD test are presented below in Table 2. P-Values suggest that the null hypothesis presented in Section 4 can be rejected in favor of the alternative hypothesis, i.e. the controller type has a statistically significant effect on the RMSR. The results of the Tukey HSD test indicate significant pairwise differences in mean RMSR between controller pairs FB/BTILC, FB/BTILC + ISC, and BTILC/BTILC + ISC.

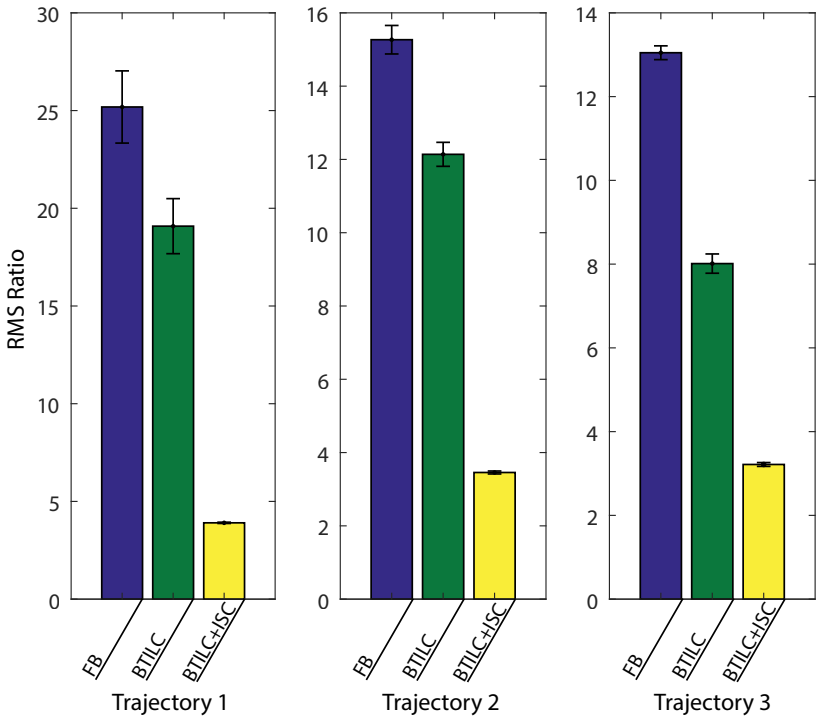

Figure 22: Tabulation of RMSR for ISC Performance Analysis

Table 2: The results of the one-way ANOVA and Tukey's HSD Test

Trajectory	P-Values	Tukey HSD
1	$< 2 \times 10^{-16}$	Significant Difference Between All Controllers
2	$< 2 \times 10^{-16}$	Significant Difference Between All Controllers
3	$< 2 \times 10^{-16}$	Significant Difference Between All Controllers

5 Discussion and Conclusion

5.1 ISC Performance Analysis

The experimental results presented in Section 4 demonstrate that the ISC provides statistically significant ($P < 0.05$) improvement in system tracking performance of control schemes employing BTILC. With an average reduction in RMSR of approximately 70%, the ISC shows a considerable level of performance improvement.

However, the ISC does display some drawbacks. Notably, oscillatory behavior appears frequently in the system output data gathered in this study when ISC is employed. In applications sensitive to vibrations, this oscillatory behavior could present a significant drawback to the ISC. The exact cause of the oscillations are unknown but similar behavior is present in each control scheme which employs the ILC generated FF inputs. This indicates that the ISC may not be the root cause of the oscillations, but a more thorough investigation should be performed on the subject.

5.2 Significance

This work represents a significant improvement in the applicability of BTILC, and ILC in general, to real-world control scenarios. Currently, ILC is limited by a cumbersome training process which often outweighs the benefits ILC provides. This is particularly evident in manufacturing environments where new system behaviors are frequently implemented. BTILC attempts to address this by adding flexibility to ILC. However, BTILC often exhibits lower performance than ILC. Adding an ISC to BTILC allows increases BTILC performance while maintaining its flexibility. In an ideal scenario, the performance of BTILC with an ISC would approach that of pure ILC.

The ISC has been posed such that it is applicable to any control scheme employing multiple exogenous FF inputs, it may produce similar performance benefits in those applications as well.

5.3 Future Work

Several questions remain for future analysis. First, the proportional gain, α, of ISC was heuristically tuned to produce acceptable results. Some effort should be applied towards explicit stability analysis and optimization of systems under ISC. Second, ISC currently functions as a proportional feedback controller; the possibility of further improving ISC by replacing α with a controller containing dynamics should be investigated. Finally, under ISC application the necessity of the system's original feedback controller is questionable as a new feedback loop has been introduced. It may be possible to completely remove the reference signal and PID existing controller from the system.

6 Acknowledgements

I'd like to thank all of you who have helped make throughout my undergraduate research experience. Including Mindy Lake for her constant encouragement, life advice, and assistance with statistical analysis; Andrej Simeunovic for his insights into mRD operation; and of course my advisor, Dr. David Hoelzle, for his consistent guidance and encouragement.

Of course, there are many others who have played a non-trivial role in my success in this project. In particular I'd like to thank Dr. Robert Siston for his support in ME 4999H, Dr. Krishnaswamy Srinivasan for serving on my defense committee, all of my friends and colleagues in the Hoelzle Research Lab, and the College of Engineering for their generous financial support.

References

[1] D. Bristow and A. Alleyne, "A high precision motion control system with application to microscale robotic deposition," *IEEE Transactions on Control Systems Technology*, vol. 16, no. 6, pp. 1008–1020, 2006.

[2] D. Hoelzle, A. Johnson, and A. Alleyne, "Bumpless transfer filter for exogenous feedforward signals," *IEEE Transactions on Control Systems Technology*, vol. 22, no. 4, pp. 1581–1588, 2014.

[3] C. Chen, *Linear System Theory and Design*, ser. Linear System Theory and Design. Oxford University Press, 1999. [Online]. Available: https://books.google.com/books?id=Fqi4oRgmOocC

[4] D. J. Hoelzle, A. G. Alleyne, and A. J. Wagoner Johnson, "Basis task approach to Iterative Learning Control with applications to micro-Robotic Deposition," *IEEE Transactions on Control Systems Technology*, vol. 19, no. 5, pp. 1138–1148, 2011.

[5] R. E. Kalman, "A new approach to linear filtering and prediction problems," *ASME Journal of Basic Engineering*, 1960.

[6] M. R. Rajamani and J. B. Rawlings, "Estimation of the disturbance structure from data using semidefinite programming and optimal weighting," *Automatica*, vol. 45, no. 1, pp. 142 – 148, 2009. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S000510980800366X

[7] D. Bristow, M. Tharayil, and A. Alleyne, "A survey of Iterative Learning Control," *Control Systems Magazine*, vol. 26, no. 3, pp. 96–114, 2006.

A Experimental Design Parameters

A.1 Plant Models

The chosen plant models for the x and y axis are

$$\mathbf{G_x}(s) = \frac{15}{s(s+1)} \tag{23}$$

$$\mathbf{G_y}(s) = \frac{25}{s(s+1)} \tag{24}$$

A.2 PID Controllers

Table 3: The PID controller gains selected for $\mathbf{C}$

Axis	k_P	k_I	k_D
x	0.075	0.200	0.100
y	0.075	0.200	0.100

A.3 Kalman Filter

The measurement error and process error covariances, R and Q, are presented below.

$$R = 6.313 \times 10^{-13} \tag{25}$$

$$Q = \begin{bmatrix} 2.889 \times 10^{-6} & -1.977 \times 10^{-8} & -2.734 \times 10^{-9} & 1.0151 \times 10^{-8} \\ -1.977 \times 10^{-8} & 2.982 \times 10^{-7} & -3.573 \times 10^{-7} & 6.527 \times 10^{-7} \\ -2.7341 \times 10^{-9} & -3.573 \times 10^{-7} & 4.282 \times 10^{-7} & -7.824 \times 10^{-7} \\ 1.015 \times 10^{-8} & 6.527 \times 10^{-7} & -7.824 \times 10^{-7} & 1.429 \times 10^{-6} \end{bmatrix} \tag{26}$$

A.4 Iterative Learning Control

The parameters for the PD-ILC learning algorithm are presented below for each axis.

Table 4: The parameters used in the x-axis ILC training process

Item	Value
k_P	0.1
k_D	15
Q Bandwidth (Hz)	35

Table 5: The parameters used in the y-axis ILC training process

Item	Value
k_P	0.1
k_D	20
Q Bandwidth (Hz)	30

A.5 Informed State Correction

The proportional gain the ISC, α, for each axis, α_x and α_y, were heuristically tuned to be

$$\boldsymbol{\alpha}_x = \begin{bmatrix} 0 & 0 \\ 0 & 1.5 \times 10^{-4} \end{bmatrix} \tag{27}$$

$$\boldsymbol{\alpha}_y = \begin{bmatrix} 0 & 0 \\ 0 & 0.5 \times 10^{-4} \end{bmatrix} \tag{28}$$